# Buggy Rating Test Approach

It is recommended that we use an exploratory testing approach. This approach allows testers to explore, learn, and analyze the website based on their intuition and insights.

When the tester tests, they will learn the website’s service and how the customer might use and value certain functions and features. In this approach, the tester will learn about the ways in which the website could fail, its weaknesses & strengths, and then they can design and execute tests on the website and report the defect

## **Parts of** Exploratory Testing

There will be three parts to this approach:

### Prepare with a Charter

To structure the exploratory testing, Session-based testing will be used. A session is a fixed duration uninterrupted testing block where the tester focuses on a particular module, feature, or scenario. The session has a deadline of 90 minutes, during which no interruptions like emails, meetings, or telephone calls are allowed. During the session, testers document various information about their testing in what’s called a charter document.

A charter document is a document that contains all the details about the session: the goal of the session, setup, notes containing helpful information along with test ideas and observations, issues uncovered during the session, and any screenshots. With this document, everyone knows the details about the session. The test charter should suggest what to test, how it can be tested, and what needs to be looked at. Test ideas are the starting point of exploration testing. The core idea here is to put some structure and direction around what to test in a timebox.

Below is a sample template of the Chater Document

|  |  |
| --- | --- |
| **Test Charter** | User Registration |
| **Actor** | new user |
| **Purpose** | To evaluate the register form of the website |
| **Setup** | A compute (desktop or Laptop), Internet Explorer version 7, Opera version 5, Mozilla……etc. (the setup might be common for several charters and can therefore be described and referred to instead of repeating the same information in every charter |
| **Priority** |  |
| **Reference** |  |
| **Data** |  |
| **Test Ideas** | 1. Verify that the user can register, and all required/mandatory fields are marked 2. Check that not filling the mandatory fields or entering blank spaces on mandatory fields and clicking the submit button will lead to a validation error. 3. Verify that validation should be in place whenever possible, and entering invalid input will lead to a validation error. 4. Verify that clicking submits button after entering all the required fields submits the data to the server. 5. After entering all the required fields, check that clicking the cancel/reset button cancels the submit request and reset all the fields. |
| **Observation** |  |

### Execute testing and keep a log

During the testing of each test case, expected outcomes, actual outcomes, and observations are logged to keep track of what was tested and what actual results occurred, how these compare to the expectation leading to the observations and anomalies if applicable in a test log.

The test log gives details about the charter session remaining or done, session setup time, scenario tested, about the testing process, a list of bugs and the issues found, and other information for the metrics.

### Discuss results, conclusions, and advice in a debriefing

At the end of the test session, the testers have a debriefing with Test Lead/Manager to review the test session findings.

The main goal of the debriefing is to convey all information that the stakeholder needs and establish their level of confidence that the pursued business value can be achieved. Preferably, the stakeholder assists the team in the debriefing by asking critical questions about their experiences during the test session.

## Exploratory Automated Testing

In addition, we will incorporate automated exploratory testing. Exploratory Automated Testing is a method integrating Test Automation within the Exploratory Testing Session that enables the testers better bug reproduction, regression tests execution, and evidence gathering. For this, it recommends using the Active Exploratory Automated Testing method

Active Exploratory Automated Testing (EAT) integrates with the Session Execution. The implementation of the Active EAT is recommended for pair testing. Usually, the first tester is responsible for creating the automated test script, while the second is responsible for the execution.

The creation of automation test scripts in this approach takes a different path than in conventional testing. Automated test scripts are made during testing and what has been discovered in the previous tests determines their design.

A closure phase is executed at the end of the EAT session. And it should have the following tasks:

* Testers involved should swap roles so that the testing resource who created the test script has a chance to re-execute the scripts to confirm the reliability & robustness of the created suite.
* A brief description along with few identifying characteristics should be provided for every automated test script.
* A criterion needs to be defined to identify which Automated test scripts can be used for the Regression test.